

Conceptos POO para Android

Este ejemplo ilustra perfectamente los pilares de la Programación Orientada a Objetos (POO) que necesitarás para Android:

1. Clase Abstracta (Empleado)

- Define comportamiento común para todos los empleados.
- Tiene el método abstracto calcularSalario() que cada subclase implementa de forma diferente.
- Comparte métodos concretos como mostrarInformacion().

2. Interfaces (Bonifiable, GestionProyectos)

- Definen contratos que las clases deben cumplir.
- Permiten múltiple herencia de comportamiento.
- Son similares a cómo en Android implementarás OnClickListener, TextWatcher, etc.

3. Herencia y Polimorfismo

- Tres tipos de empleados heredan de Empleado.
- Cada uno implementa calcularSalario() de forma distinta.
- Puedes tratar a todos como Empleado dentro de arrays o listas.

4. Encapsulación

- Uso de atributos protected y private.
- Acceso controlado mediante getters.

5. ¿Por qué es perfecto para Android?

En Android utilizarás estos mismos conceptos:

- Clases abstractas: Activity, Fragment, View
- Interfaces: OnClickListener, TextWatcher, Callback
- Herencia: Extenderás AppCompatActivity, RecyclerView.Adapter
- Polimorfismo: Manejarás diferentes vistas de forma uniforme.

